

(12)

EUROPEAN PATENT APPLICATION

(21) Application number: 79103408.5

(51) Int. Cl.³: **G 06 F 3/04**

(22) Date of filing: 12.09.79

(30) Priority: 02.10.78 US 948070

(43) Date of publication of application:
16.04.80 Bulletin 80/8

(84) Designated Contracting States:
BE CH DE FR GB IT NL SE

(71) Applicant: International Business Machines Corporation

Armonk, N.Y. 10504(US)

(72) Inventor: Adams Jr., Robert Lowell
33 Amsterdam Avenue
Kingston New York 12401(US)

(72) Inventor: Grant, Carl Henry
20 Niles Drive
Woodstock New York 12498(US)

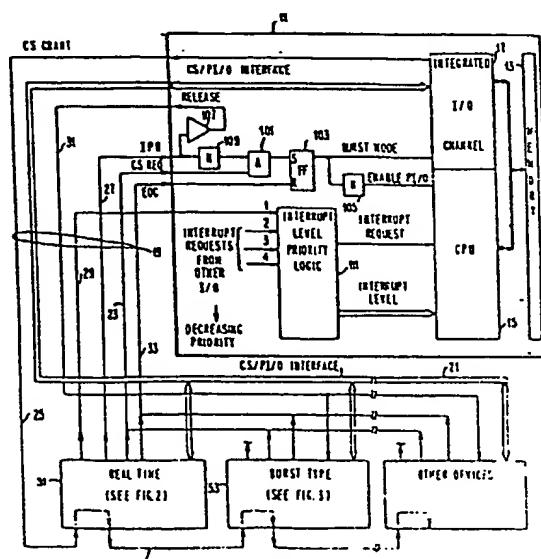
(72) Inventor: Stevens, Karl Wayhe
7004 Latham Circle
Saugerties New York 12477(US)

(74) Representative: Richards, John Peter
IBM UNITED KINGDOM PATENT OPERATIONS Hursley Park
Winchester Hants SO21 2JN(GB)

(54) Computer input/output apparatus and method.

(57) The invention is directed to an improved computer I/O interrupt sequencing method and apparatus for the generation of an instruction priority request signal to a burst mode device operating in a direct memory access mode in order to indicate that a real time task requires programmed I/O service. An end of chain signal is generated by the burst mode device receiving the instruction priority request signal, thereby suspending the burst I/O control of the I/O bus and allowing programmed I/O service to a real time device. After the needs of the real time device have been satisfied, the adapter of the real time device removes the instruction priority request signal, thereby allowing the burst mode data transfer to continue at the count positions at which it was suspended.

EP 0 009 678 A1



- 1 -

COMPUTER INPUT/OUTPUT APPARATUS AND METHOD

The present invention generally relates to data processing systems and more particularly to input/output control associated with real time devices operating on a general purpose data processing system.

At least two types of input/output operations are known in the prior art. In a first type, an I/O program is executed within a computer going through the steps of addressing an I/O device, sending a command to the addressed I/O device to instruct the device to either send or receive data, and either receiving or sending several bytes of data to the I/O device. Usually, the amount of data transferred during one I/O device selection is kept small so that the channel will be available without long delays for receiving interrupt requests from other transfer devices requiring service. During program input/output data transfer, the data passes through the Central Processing Unit (CPU) and is stored in memory by the CPU instruction. Real time devices often use the programmed input/output data transfer method in order that the computer may have immediate access to the data being received. Immediate data access allows immediate response calculation for transmission to the real time device.

Nonreal time devices such as disk memories and tape drives can be more efficiently operated in what is sometimes termed "device initiated burst mode". Device

- 2 -

initiated data transfer is initiated by a signal such as cycle steal request being sent from a device to a computer which terminates instruction execution after completion of the currently executing instruction. Upon termination of instruction execution, the computer sends a cycle steal grant response to all I/O devices which are connected in a daisy chain sequence. If two devices require service simultaneously, both will raise a cycle steal request signal but the device with highest priority will receive the cycle steal grant signal first and can inhibit propagation to the lower priority device. Having received the cycle steal grant signal, the disk control logic can place an identifying port address on the data bus, a command indicating whether the device which is to send or receive data and thereafter a single sequential burst of data bytes without intervening address and command information. When in cycle steal mode, data does not pass through the CPU itself, but rather is passed directly to sequential memory addresses under control of an indexing address counter in the selected port of the input/output channel. In addition to the address counter, each port has a byte counter in the associated burst mode device containing the length of the data transfer. As each byte is transferred, the count is decremented. When the count reaches zero, the data transfer has been completed and the burst mode device generates an end of message signal. Each port address counter and byte counter are loaded by programmed instructions prior to the start of any burst data transfer. If the input/output channel is slower than memory, time slots may become available within which the computer can be given

- 3 -

access to the memory without interfering with data being transferred by the channel.

It should be noted however, that in spite of the fact that the Central Processor Unit may operate in memory, the Central Processor Unit cannot execute a programmed I/O operation to serve a real time device until the entire burst of data has been transmitted. The burst of data may constitute a single block of 256 or 1,024 bytes requiring significant transmission time and leading to excessive response time for the real time device.

Breaking up the burst into a series of short bursts may allow adequate service to the real time device but seriously impacts burst mode efficiency since the cycle steal request-cycle steal grant sequence must be repeated far more often with small bursts.

A partial solution to this problem is proposed in the prior art by permitting a higher priority device capable of device initiated data transfer to suspend transfer by a lower priority device in the middle of a burst and substitute its own port address, command and/or data sequence. The lower priority device resumes the transmission of its burst upon completion of transmission by the higher priority device. The above described partial solution is incomplete because although a real time device can provide its data into memory, it has no way of receiving a response until all of the lower priority transmitting and suspended bursts have been completed. Until completed, the channel

- 4 -

interface is not available for programmed input/output operations by the CPU. Furthermore, the CPU may be unaware of the existence of the real time data in memory since programmed I/O interruptions are inhibited during burst mode channel operations.

Accordingly, it is an object of the invention to provide an improved computer input/output method and apparatus particularly suited to the needs of real time devices operating in combination with device initiated burst mode devices on a common I/O interface.

According to the present invention there is provided a computer input/output apparatus adapted to operate in either device initiated burst mode or programmed I/O mode, the apparatus being characterised by an interrupt sequencing apparatus including means in a first device attached to said input/output apparatus for generating an instruction priority request signal thereby indicating to said computer that immediate programmed I/O service is required and requesting a second device attached to said input/output apparatus to suspend a burst data transfer in progress, means in said second device for generating an end of chain signal to said computer to indicate suspension of burst data transfer; means in said computer responsive to said end of chain signal to terminate burst control of said input/ output apparatus whereby said input/output apparatus is made available for programmed I/O operation; and means responsive to a programmed I/O instruct-

- 5 -

ion for permitting device initiated burst mode data transfers to resume.

The invention further provides a method of suspending a burst mode device initiated data transfer on a computer input/output channel for programmed I/O service of a real time device characterised by the steps of generating an instruction priority request signal in said real time device, transmitting said instruction priority request signal to said burst mode device, generating an end of chain signal in said burst mode device in response to said instruction priority request, transmitting said end of chain signal to said computer, switching said input/output channel from burst mode to programmed I/O mode, and terminating said instruction priority signal.

An embodiment of the invention will now be described, by way of example, with reference to the accompanying drawings, in which:

Figure 1 is a block diagram of input/output devices connected to an input/output bus which is in turn connected to a computer and showing detailed logic within the computer,

Figure 2 is a detailed diagram showing logic within a real time device,

Figure 3 is a detail circuit diagram of logic in a burst type device, and

- 6 -

Figure 4 is a timing diagram showing a programmed I/O operation during a suspended burst I/O operation.

In Fig. 1, a computer 11, including a memory 13, a multiple interrupt level Central Processing Unit 15 and an integrated input/output channel 17 are shown connected to a common input/output bus 19. Common bus 19 includes CS/PIO interface 21 and additional lines of the invention. The interface 21 includes, for example, sixteen data wires and two parity wires for simultaneous parallel transmission of two bytes of data with parity. Interface 21 will also include synchronizing tag lines. These tag lines indicate the content of data on the data wires to be an address, a command, or data and may, for example, be labelled TA, TC and TD. TA indicates, for example, that the information on the data wires is an address, whereas TC and TD indicate that the information is a command or data. The signal lines described above are shown as a single interface cable 21, and are well known to persons skilled in the art of computer architecture. Additional signal lines 23 and 25 relate to device initiated burst mode transmission. Cycle steal request line 23 can be activated by any of the burst type devices which may also be a real time device. Cycle steal request line 23 is an input to AND gate 101 which in turn sets the flip-flop 103. Flip-flop 103 being set signals to integrated I/O channel 17 that a device initiated burst mode data transfer is being requested. When the Central Processing Unit 15 has completed execution of an instruction cycle which may be in progress, control of memory 13 is

transferred to integrated I/O channel 17 and a cycle steal grant signal is propagated on line 25 through the I/O devices in a prioritized daisy chain. The burst mode output signal from flip-flop 103 is inverted by inverter 105 to remove the enable programmed I/O signal from the Central Processing Unit 15 thereby inhibiting CPU 15 from executing programmed I/O operations which would conflict with burst mode operations.

In order to suspend burst mode operations and allow programmed I/O operations to resume, an instruction priority request signal can be generated by a real time device 51. For purposes of clarity of explanation, the instruction priority request signal is shown as being sent to computer 11 where it is amplified by amplifier 107 and sent to burst mode devices as a release signal on line 31. It will be recognized by those skilled in the art that the instruction priority request signal could have been sent directly to the burst mode devices without prior amplification by amplifier 107 in computer 11 so long as electrical drive capacities are not exceeded. In addition to being sent to burst mode devices, the instruction priority request signal is inverted by inverter 109 and applied to a second input of AND gate 101 to prevent flip-flop 103 from being set by burst mode devices after it becomes reset by an end of chain signal to be described hereafter.

- 8 -

In response to a real time device having generated an instruction priority request signal, a burst mode device 53 which may be in the midst of a data transfer burst will generate an end of chain signal on line 33 and terminate data transfer. The end of chain signal resets latch 103 which in conjunction with inverter 105 enables programmed I/O operations by Central Processing Unit 15. In conjunction with the return to programmed I/O capability, the real time device would present its interrupt request on line 29 allowing interrupt level priority logic 111 to signal CPU 15 that programmed I/O service is required by real time device 51.

Referring now to Fig. 2, simplified detail logic is shown responsive to the needs of the real time device and to the system. More particularly, flip-flop 201 has a set input which is connected to an output of a real time event detector such as a voltage comparator, a relay, etc. signalling the need for real time process- or intervention. Flip-flop 201 being set provides the instruction priority request signal previously discussed with respect to the signals on wires 27 and 31. In conjunction with the request for immediate programmed I/O service, the real time device also provides its interrupt request via flip-flop 203 to interrupt level priority logic 111. The actual interrupt to CPU 15 will be presented to the internal interrupt circuits of CPU 15 when CPU 15 regains programmed I/O control of the input/output bus 19. Each of flip-flops 201 and 203 are reset by programmed I/O commands from CPU

15 which are decoded by well-known command decode circuitry and applied to reset signal lines 205 and 207 respectively.

Referring now to Fig. 3, those additional circuits which would be added or changed within a suspendable burst type I/O device such as a disk or tape are shown. By suspendable burst type device, a buffer or other non-overrunable device is to be considered. A burst mode transfer of data is initiated by a condition such as buffer full during a read operation or buffer empty during a write operation. The buffer full or buffer empty operation is recognized as a service request which sets latch 301. Latch 301 being set causes AND gate 303 to propagate a cycle steal request signal onto line 23 of Fig. 1 because at this state active flip-flop 305 has not yet been set and accordingly inverter 307 satisfies the input conditions of AND gate 303. Having generated a cycle steal request, in due course, a cycle steal grant signal will appear and set active flip-flop 305 via AND gate 309. Inputs of AND gate 309 are connected to flip-flop 301, as well as the cycle steal grant input line. Because service request flip-flop 301 is set, inverter 311 having an output connected to AND gate 313 prevents propagation of the delayed cycle steal grant signal to the next lower priority device on bus 19. The output of active flip-flop 305 is connected to inverter 307 as well as AND gates 315 and 317. Additionally, the output of active flip-flop 305 is connected to the reset inputs of flip-flops 323, 325 and 327. AND gate 315 has an output

- 10 -

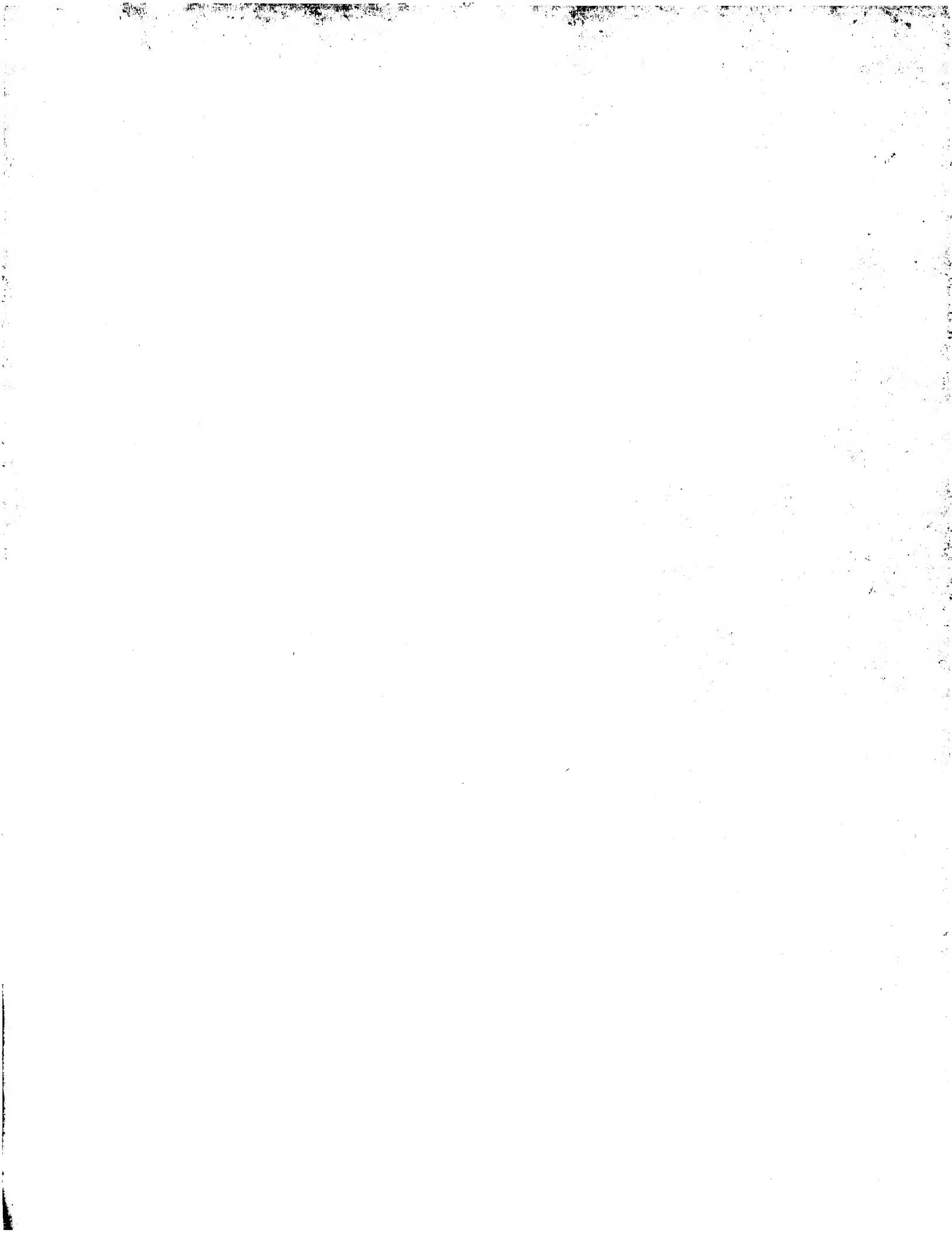
connected to OR gate 329, which is in turn connected to AND gates 319, 331, and 333 as well as inverter 321. AND gate 317 is likewise connected to OR gate 329. The outputs of AND gates 315 and 317 are connected to control and data register gates not shown to transfer control information identifying the burst device channel port number to integrated I/O channel 17, while AND gate 317 transfers data between a buffer or non-overrunnable I/O device and memory 13 at the data tag signal time TD. Accompanying each data transfer will be a valid signal on the valid tag line generated by AND gate 333, which has inputs connected to service request flip-flop 301 as well as OR gate 329. If the end of the message is detected, usually by the byte counter value equal to zero, flip-flop 301 is reset while OR gate 335 is conditioned to provide the end of chain output signal via AND gate 331. End of chain indicates to integrated I/O channel 17 that the burst data transfer has been completed. Another input to OR gate 335 receives a signal from release flip-flop 325, which is in turn set by AND gate 319, which is in turn synchronized by flip-flop 323. AND gate 337 has an input for receiving the release signal described earlier as being an amplified instruction priority request signal, which in combination with inverter 321 may set release synch flip-flop 323 between TD signals. The next TD signal sets flip-flop 325. Release latch 325 being set allows AND gate 331 to generate an end of chain signal on line 33 even though normal end of message conditions have not been reached. As previously mentioned, the end of chain signal resets latch

- 11 -

103 in computer 11 thereby terminating the burst mode operation and enabling programmed I/O operation. The end of chain signal also is fed back to set latch 327, which allows AND gate 339 with inverter 341 to reset active flip-flop 305 as soon as the data transfer tag TD signal is removed. Active latch 305 being reset prevents AND gate 315, 317 from further data transmission thereby freeing the bus 19 for use by CPU 15.

With the figures and foregoing description in mind, the operation of the preferred embodiment of the invention will now be described. Before burst mode operation can commence, CPU 15 must load the address counter and byte counter of each port to be used. The address counters are loaded with the starting address of the blocks of memory in memory 13 which are to be involved in burst data transfers through each port. Likewise, a programmed I/O operation will load the channel port number and block size count into a byte counter in each burst mode device. Thereafter as the buffer in a burst mode device becomes either empty when writing or full when reading, a service request is generated internal to the burst mode device causing a cycle steal request for data transfer. This cycle steal request sets burst mode latch 103 which results in integrated I/O channel 17 providing a cycle steal grant signal.

Referring now to Fig. 3, when the cycle steal grant signal is received by AND gate 315, the control word identifying the burst device and including the number of the channel port to be used is gated out to





Europäisches Patentamt
European Patent Office
Office européen des brevets

⑪ Publication number:

O 009 678
B1

⑫

EUROPEAN PATENT SPECIFICATION

⑯ Date of publication of patent specification: **08.12.82** ⑮ Int. Cl.³: **G 06 F 3/04**
⑯ Application number: **79103408.5**
⑯ Date of filing: **12.09.79**

④ Computer input/output apparatus.

EP O 009 678 B1

⑩ Priority: **02.10.78 US 948070**
⑯ Date of publication of application:
16.04.80 Bulletin 80/8
⑯ Publication of the grant of the patent:
08.12.82 Bulletin 82/49
⑯ Designated Contracting States:
BE CH DE FR GB IT NL SE
⑯ References cited:
IBM TECHNICAL DISCLOSURE BULLETIN, vol.
17, no. 9, February 1975.
New York (US)
MILLER et al.: "Interleaved Cycle Steal Burst
Circuitry", pages 2732—2735.

IBM TECHNICAL DISCLOSURE BULLETIN, vol.
17, no. 10, March 1975
New York (US)
BODNER et al.: "Multiple Class Cycle Steal
System", Pages 2992—3001.

⑬ Proprietor: **International Business Machines
Corporation
Armonk, N.Y. 10504 (US)**

⑭ Inventor: **Adams Jr., Robert Lowell
33 Amsterdam Avenue
Kingston New York 12401 (US)**
Inventor: **Grant, Carl Henry
20 Niles Drive
Woodstock New York 12498 (US)**
Inventor: **Stevens, Karl Wayhe
7004 Latham Circle
Saugerties New York 12477 (US)**

⑮ Representative: **Richards, John Peter
IBM UNITED KINGDOM PATENT OPERATIONS
Hursley Park
Winchester Hants, SO21 2JN (GB)**

Note: Within nine months from the publication of the mention of the grant of the European patent, any person may give notice to the European Patent Office of opposition to the European patent granted. Notice of opposition shall be filed in a written reasoned statement. It shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European patent convention).

Computer input/output apparatus

The present invention relates to a computer input/output apparatus as indicated in the pre-characterising part of claim 1. Such apparatus is known from IBM Technical Disclosure Bulletin, vol. 17, no. 10, March 1975, pp. 2992 to 3001.

At least two types of input/output operations are known in the prior art. In a first type, an I/O program is executed within a computer going through the steps of addressing an I/O device, sending a command to the addressed I/O device to instruct the device to either send or receive data, and either receiving or sending several bytes of data to the I/O device. Usually, the amount of data transferred during one I/O device selection is kept small so that the channel will be available without long delays for receiving interrupt requests from other transfer devices requiring service. During program input/output data transfer, the data passes through the Central Processing Unit (CPU) and is stored in the memory by the CPU instruction. Real time devices often use the programmed input/output data transfer method in order that the computer may have immediate access to the data being received. Immediate data access allows immediate response calculation for transmission to the real time device.

Nonreal time devices such as disk memories and tape drives can be more efficiently operated in what is sometimes termed "device initiated burst mode". Device initiated data transfer is initiated by a signal such as cycle steal request being sent from a device to a computer which terminates instruction execution after completion of the currently executing instruction. Upon termination of instruction execution, the computer sends a cycle steal grant response to all I/O devices which are connected in a daisy chain sequence. If two devices require service simultaneously, both will raise a cycle steal request signal but the device with highest priority will receive the cycle steal grant signal first and can inhibit propagation to the lower priority device. Having received the cycle steal grant signal, the disk control logic can place an identifying port address on the data bus, a command indicating whether the device which is to send or receive data and thereafter a single sequential burst of data bytes without intervening address and command information. When in cycle steal mode, data does not pass through the CPU itself, but rather is passed directly to sequential memory addresses under control of an indexing address counter in the selected port of the input/output channel. In addition to the address counter, each port has a byte counter in the associated burst mode device containing the length of the data transfer. As each byte is transferred, the count is decremented. When the count reaches zero, the data transfer has been completed and the burst mode device

generates an end of chain signal. Each port address counter and byte counter are loaded by programmed instructions prior to the start of any burst data transfer. If the input/output channel is slower than memory, time slots may become available within which the computer can be given access to the memory without interfering with data being transferred by the channel.

It should be noted however, that in spite of the fact that the Central Processor Unit may operate in memory, the Central processor Unit cannot execute a programmed I/O operation to serve a real time device until the entire burst of data has been transmitted. The burst of data may constitute a single block of 256 or 1,024 bytes requiring significant transmission time and leading to excessive response time for the real time device.

Breaking up the burst into a series of short bursts may allow adequate service to the real time device but seriously impacts burst mode efficiency since the cycle steal request-cycle steal grant sequence must be repeated far more often with small bursts.

A partial solution to this problem is proposed in the prior art by permitting a higher priority device capable of device initiated data transfer to suspend transfer by a lower priority device in the middle of a burst and substitute its own port address, command and/or data sequence. The lower priority device resumes the transmission of its burst upon completion of transmission by the high priority device. The above described partial solution is incomplete because although a real time device can provide its data into memory, it has no way of receiving a response until all of the lower priority transmitting and suspended bursts have been completed. Until completed, the channel interface is not available for programmed input/output operations by the CPU. Furthermore, the CPU may be unaware of the existence of the real time data in memory since programmed I/O interruptions are inhibited during burst mode channel operations.

Accordingly, it is an object of the invention to provide an improved computer input/output method and apparatus particularly suited to the needs of real time devices operating in combination with device initiated burst mode devices on a common I/O interface.

According to the present there is provided a computer input/output apparatus which comprises the features set out in claim 1.

Particular embodiments of the invention are set out in the subsidiary claims.

An embodiment of the invention will now be described, by way of example, with reference to the accompanying drawings, in which:

Figure 1 is a block diagram of input/output devices connected to an input/output bus which is in turn connected to a computer and show-

ing detailed logic within the computer.

Figure 2 is a detailed diagram showing logic within a real time device.

Figure 3 is a detail circuit diagram of logic in a burst type device, and

Figure 4 is a timing diagram showing a programmed I/O operation during a suspended burst I/O operation.

In Fig. 1, a computer 11, including a memory 13, a multiple interrupt level Central Processing Unit 15 and an integrated input/output channel 17 are shown connected to a common input/output bus 19. Common bus 19 includes CS/PIO interface 21 and additional lines of the invention. The interface 21 includes, for example, sixteen data wires and two parity wires for simultaneous parallel transmission of two bytes of data with parity. Interface 21 will also include synchronizing tag lines. These tag lines indicate the content of data on the data wires to be an address, a command, or data and may, for example, be labelled TA, TC and TD. TA indicates, for example that the information on the data wires is an address, whereas TC and TD indicate that the information is a command or data. The signal lines described above are shown as a single interface cable 21, and are well known to persons skilled in the art of computer architecture. Additional signal lines 23 and 25 relate to device initiated burst mode transmission. Cycle steal request line 23 can be activated by any of the burst type devices which may also be a real time device. Cycle steal request line 23 is an input to AND gate 101 which in turn sets the flip-flop 103. Flip-flop 103 being set signals to integrated I/O channel 17 that a device initiated burst mode data transfer is being requested. When the Central Processing Unit 15 has completed execution of an instruction cycle which may be in progress, control of memory 13 is transferred to integrated I/O channel 17 and a cycle steal grant signal is propagated on line 25 through the I/O devices in a prioritized daisy chain. The burst mode output signal from flip-flop 103 is inverted by inverter 105 to remove the enable programmed I/O signal from the Central Processing Unit 15 thereby inhibiting CPU 15 from executing programmed I/O operations which would conflict with burst mode operations.

In order to suspend burst mode operations and allow programmed I/O operations to resume, an instruction priority request signal can be generated by a real time device 51. For purposes of clarity of explanation, the instruction priority request signal is shown as being sent to computer 11 where it is amplified by amplifier 107 and sent to burst mode devices as a release signal on line 31. It will be recognized by those skilled in the art that the instruction priority request signal could have been sent directly to the burst mode devices without prior amplification by amplifier 107 in computer 11 so long as electrical drive capacities are not exceeded. In addition to being sent to burst mode

devices, the instruction priority request signal is inverted by inverter 109 and applied to a second input of AND gate 101 to prevent flip-flop 103 from being set by burst mode devices after it becomes reset by a forced end of chain signal to be described hereafter.

In response to a real time device having generated an instruction priority request signal, a burst mode device 53 which may be in the midst of a data transfer burst will generate the forced end of chain signal on line 33 and terminate data transfer. The forced end of chain signal resets latch 103 which in conjunction with inverter 105 enables programmed I/O operations by Central Processing Unit 15. In conjunction with the return to programmed I/O capability, the real time device would present its interrupt request on line 29 allowing interrupt level priority logic 111 to signal CPU 15 that programmed I/O service is required by real time device 51.

Referring now to Fig. 2, simplified detail logic is shown responsive to the needs of the real time device and to the system. More particularly, flip-flop 201 has a set input which is connected to an output of a real time event detector such as a voltage comparator, a relay, etc. signalling the need for real time processor intervention. Flip-flop 201 being set provides the instruction priority request signal previously discussed with respect to the signals on wires 27 and 31. In conjunction with the request for immediate programmed I/O service, the real time device also provides its interrupt request via flip-flop 203 to interrupt level priority logic 111. The actual interrupt to CPU 15 will be presented to the internal interrupt circuits of CPU 15 when CPU 15 regains programmed I/O control of the input/output bus 19. Each of flip-flops 201 and 203 are reset by programmed I/O commands from CPU 15 which are decoded by well-known command decode circuitry and applied to reset signal lines 205 and 207 respectively.

Referring now to Fig. 3, those additional circuits which would be added or changed within a suspendable burst type I/O device such as a disk or tape are shown. By suspendable burst type device, a buffer or other non-overrunable device is to be considered. A burst mode transfer of data is initiated by a condition such as buffer full during a read operation or buffer empty during a write operation. The buffer full or buffer empty operation is recognized as a service request which sets latch 301. Latch 301 being set causes AND gate 303 to propagate a cycle steal request signal onto line 23 of Fig. 1 because at this state active flip-flop 305 has not yet been set and accordingly inverter 307 satisfies the input conditions of AND gate 303. Having generated a cycle steal request, in due course, a cycle steal grant signal will appear and set active flip-flop 305 via AND gate 309. Inputs of AND gate 309 are connected to flip-flop 301, as well as the cycle steal grant input line. Because

service request flip-flop 301 is set, inverter 311 having an output connected to AND gate 313 prevents propagation of the delayed cycle steal grant signal to the next lower priority device on bus 19. The output of active flip-flop 305 is connected to inverter 307 as well as AND gates 315 and 317. Additionally, the output of active flip-flop 305 is connected to the reset inputs of flip-flops 323, 325 and 327. AND gate 315 has an output connected to OR gate 329, which is in turn connected to AND gates 319, 331, and 333 as well as inverter 321. AND gate 317 is likewise connected to OR gate 329. The outputs of AND gates 315 and 317 are connected to control and data register gates not shown to transfer control information identifying the burst device channel port number to integrated I/O channel 17, while AND gate 317 transfers data between a buffer or non-overrunable I/O device and memory 13 at the data tag signal time TD. Accompanying each data transfer will be a valid signal on the valid tag line generated by AND gate 333, which has inputs connected to service request flip-flop 301 as well as OR gate 329. If the end of the message is detected, usually by the byte counter value equal to zero, flip-flop 301 is reset while OR gate 335 is conditioned to provide an end of chain signal via AND gate 331. The end of chain signal indicates to integrated I/O channel 17 that the burst data transfer has been completed. Another input to OR gate 335 receives a signal from release flip-flop 325, which is in turn set by AND gate 319, which is in turn synchronized by flip-flop 323. AND gate 337 has an input for receiving the release signal described earlier as being an amplified instruction priority request signal, which in combination with inverter 321 may set release sync flip-flop 323 between TD signals. The next TD signal sets flip-flop 325. Release latch 325 being set allows AND gate 331 to generate an end of chain signal on line 33 even though normal end of message conditions have not been reached. Such a signal is referred to herein as a forced end of chain signal to distinguish it from a true end of chain signal signifying the actual completion of burst data transfer. The forced end of chain signal resets latch 103 in computer 11 thereby terminating the burst mode operation and enabling programmed I/O operation. The forced end of chain signal also is fed back to set latch 327, which allows AND gate 339 with inverter 341 to reset active flip-flop 305 as soon as the data transfer tag TD signal is removed. Active latch 305 being reset prevents AND gate 315, 317 from further data transmission thereby freeing the bus 19 for use by CPU 15.

With the figures and foregoing description in mind, the operation of the preferred embodiment of the invention will now be described. Before burst mode operation can commence, CPU 15 must load the address counter and byte counter of each port to be used. The address counters are loaded with the starting address of

the blocks of memory in memory 13 which are to be involved in burst data transfers through each port. Likewise, a programmed I/O operation will load the channel port number and block size counter into a byte counter in each burst mode device. Thereafter as the buffer in a burst mode device becomes either empty when writing or full when reading, a service request is generated internal to the burst mode device causing a cycle steal request for data transfer. This cycle steal request sets burst mode latch 103 which results in integrated I/O channel 17 providing a cycle steal grant signal.

Referring now to Fig. 3, when the cycle steal grant signal is received by AND gate 315, the control word identifying the burst device and including the number of the channel port to be used is gated out to the data bus along with a valid tag signal generated by AND gate 333. Integrated I/O channel 17 of Fig. 1 uses the port number to identify the address counter containing the memory addresses involved in the transfer. Thereafter a continuous sequence of TD data timing pulses, each associated with two bytes of data on the data wires of I/O bus 19. Each TD signal control AND gate 317 of Fig. 3 to pass the data either into or out of burst device 53 and to increment the data counter. When the buffer has been emptied during a read operation or filled during write operation, an end of message is provided to reset service register 301 and generate an end of chain signal through OR gate 335 via AND gate 331.

The foregoing description followed the operation of the circuits of the preferred embodiment while transmitting a single uninterrupted burst of data.

When a real time device 51 requires service, the latches 201 and 203 in Fig. 2 are set by the real time event detected within the device. Latches 201 and 203 generate instruction priority request and interrupt request signals respectively on lines 27 and 23 of bus 19. The instruction priority request signal is inverted by inverter 109 to inhibit AND gate 101 and thereby prevent setting burst mode latch 103 if it has not already been set. In the event that it is already set, it will be reset by the forced end of chain signal from the burst transferring device.

Referring now to Fig. 4, the sequence of an instruction priority request causing a programmed I/O sequence to suspend an executing burst mode data transfer will be described with the other figures in mind. As shown in the first waveform, instruction priority request (IPR) 401 is raised asynchronously when the real time device first requires service. Having raised the instruction priority request line, the release input via AND gate 337 sets release sync latch 323 at the end of the TD pulse. Latch 323 being set in turn causes release latch 325 to be set at the beginning of the next TD pulse. The output of release latch 325 forces an end of chain signal 403 via gates 335 and 331. The forced end of chain signal

from AND gate 331 causes flip-flop 327 to be set which in turns resets active flip-flop 305 when the end of chain signal falls with TD. Active latch 305 being reset allows inverter 307 to co-operate with service request register 301 which remains set to create a cycle steal request 405 which will remain active throughout the following programmed I/O sequence and re-establish the burst data transfer thereafter.

Burst data transfer is reestablished when the programmed I/O instruction resets flip-flop 201 thereby removing the instruction priority request signal 407 and allowing a cycle steal grant 409 to be generated by integrated I/O channel 17. The cycle steal grant signal anded with the output of service register flip-flop 301 at AND gate 309 to again set active flip-flop 305. Active flip-flop 305 being set in combination with the cycle steal grant input signal causes the control word 411 identifying the port being used to again be gated out on the data lines. When the control word is received by integrated I/O channel 17, it drops the signal on the cycle steal grant line and begins transmitting data timing pulses TD which are thereafter associated with each pair of data bytes 413 transferred.

While the invention has been particularly shown and described with reference to a preferred embodiment thereof including an integrated channel, it will be understood by those skilled in the art that various changes in form and detail including use of a separate I/O channel or the substitution of parallel cycle steal grant priority determination logic may be made therein.

Claims

1. A computer input/output (I/O) apparatus adapted to operate in either a programmed I/O mode initiated by a first device or a burst mode initiated by a second device, the apparatus comprising an interrupt sequencing means normally giving priority to the burst mode, and the second device including means for generating a signal to indicate suspension of burst data transfer and means for permitting device initiated burst mode data transfer to resume, the apparatus being characterised by means in the first device for generating a programmed instruction priority request signal thereby indicating to the computer that immediate programmed I/O service is required and requesting the second device to suspend a burst data transfer in progress, means in the second device for generating a forced end of chain signal to the computer to indicate suspension of burst data transfer, means in the computer responsive to the forced end of chain signal to terminate burst control of the input/output apparatus whereby the input/output apparatus is made available for programmed I/O operation, and means responsive to a programmed I/O instruction for permitting device initiated

5 burst mode data transfers to resume.

2. The input/output apparatus of Claim 1 wherein the computer further comprises means responsive to the instruction priority request signal to inhibit initiation of a higher priority burst mode data transfer.

10 3. The input/output apparatus of Claim 1 wherein the means responsive to a programmed I/O instruction further comprises means in the first device responsive to the programmed I/O instruction to terminate the instruction priority request signal, and means in the computer responsive to the termination of the instruction priority signal for generating a grant signal to the second device permitting the second device to resume burst mode data transfer.

Revendications

20 1. Système d'entrée/sortie pour ordinateur pouvant fonctionner soit dans un mode dit de transfert programmé de données d'entrée/sortie qui est initialisé par un premier dispositif, soit dans un mode dit de transfert de données en rafales qui est initialisé par un second dispositif, ledit système comprenant des moyens de mise en séquence des interruptions, lesquels donnent normalement la priorité audit mode de transfert de données en rafales, le second dispositif comprenant des moyens permettant d'engendrer un signal indiquant la suspension des transferts de données en rafales, et des moyens permettant la reprise, déclenchée par le dispositif, desdits transferts de données; le système étant caractérisé en ce que ledit premier dispositif comprend des moyens permettant d'engendrer un signal de demande d'attribution de priorité programmée, pour informer ainsi l'ordinateur qu'un passage immédiat au mode dit de transfert programmé de données d'entrée/sortie est nécessaire et demander audit second dispositif d'interrompre les transferts de données en rafales en cours; en ce que ledit second dispositif comporte des moyens permettant d'engendrer un signal de cessation forcée des transferts de données en rafales et de transmettre celui-ci à l'ordinateur pour indiquer l'interruption des transferts de données en rafales; en ce que l'ordinateur comporte des moyens permettant, en réponse à ce dernier signal, de mettre fin au fonctionnement du système dans ledit mode de transfert de données en rafales, le système étant alors disponible aux fins d'opérations d'entrée/sortie programmées; et en ce que le système comprend des moyens permettant, en réponse à une instruction d'entrée/sortie programmée, la reprise, déclenchée par le dispositif, des transferts de données en rafales.

25 40 45 50 55 60 65 2. Système d'entrée/sortie selon la revendication 1, caractérisé en ce que l'ordinateur comprend en outre des moyens permettant, en réponse audit signal de demande d'attribution de priorité, d'interdire l'initialisation d'un trans-

fert de données en rafales présentant une priorité plus élevée.

3. Système d'entrée/sortie selon la revendication 1, caractérisé en ce que les moyens répondant à une instruction d'entrée/sortie programmée comprennent en outre des moyens situés dans ledit premier dispositif et permettant, en réponse à l'instruction d'entrée/sortie programmée, de mettre fin audit signal de demande d'attribution de priorité, et des moyens situés dans l'ordinateur qui lui permettent, lors de la cessation de ce dernier signal, d'engendrer un signal autorisant ledit second dispositif à reprendre les transferts de données en rafales.

Patentansprüche

1. Ein- und Ausgabe-Steuereinheit eines Rechnersystems, die entweder im programmierten Ein- und Ausgabe-Betrieb, der von einer ersten Ein- und Ausgabeeinheit initiiert wird oder im Einpunktbetrieb, der von einer zweiten Ein- und Ausgabeeinheit initiiert wird, arbeitet, wobei die Ein- und Ausgabesteuereinheit eine Unterbrechungsfolgeschaltung aufweist, die normalerweise dem Einpunktbetrieb die Priorität zuordnet, und die zweite Ein- und Ausgabeeinheit eine Schaltung zur Erzeugung eines Signals enthält, um das Suspendieren der Datenübertragung im Einpunktbetrieb anzuzeigen und weiterhin eine Schaltung enthält, die es ermöglicht, das die Einheit, die den Einpunkt-Datentransfer initiiert hat, diesen weiterführen kann, dadurch gekennzeichnet, daß in der ersten Einheit eine Schaltungsanordnung vorhanden ist, die ein programmiertes Instruktions-Prioritäts-Anforderungssignal erzeugt, wodurch dem Rechner Angezeigt wird, daß eine sofortige programmierte Ein- und Ausgabebedienung erforderlich ist und die die zweite

Einheit anweist, die Einpunkt-Datenübertragung zu suspendieren, daß außerdem in der zweiten Einheit Schaltungen vorhanden sind, die ein erzwungenes Ende des Kettensignals erzeugen, um dem Rechner anzuzeigen, daß die Einpunkt-Datenübertragung suspendiert ist, daß im Rechner Schaltungsanordnungen vorhanden sind, die auf das erzwungene Ende des Kettensignals ansprechen, um die Einpunkt-Steuerung der Ein- und Ausgabesteuereinheit zu beenden, wodurch diese in dem programmierten Ein- und Ausgabe-Betrieb umgeschaltet wird, und daß Schaltungen vorhanden sind, die auf eine programmierte Ein- und Ausgabeeinheit ansprechen, damit die Ein- und Ausgabeeinheit, die die Einpunkt-Datenübertragung initiiert hat, den Einpunktbetrieb wieder aufzunehmen kann.

2. Ein- und Ausgabesteuereinheit nach Anspruch 1, dadurch gekennzeichnet, daß der Rechner außerdem Schaltungen Enthält, die auf ein Instruktions-Prioritäts-Anforderungssignal ansprechen, um die Initiierung einer Einpunkt-Datenübertragung mit höherer Priorität zu verhindern.

3. Ein- und Ausgabesteuereinheit nach Anspruch 1, dadurch gekennzeichnet, daß die Schaltungen, die auf eine programmierte Ein- und Ausgabeanstruktion ansprechen, weitere Schaltungen in der ersten Ein- und Ausgabeeinheit enthalten, die auf die programmierte Ein- und Ausgabeanstruktion ansprechen, um das Instruktions-Prioritäts-Anforderungssignal zu beenden, und daß im Rechner Schaltungen vorhanden sind, die auf das Beenden des Instruktionsprioritätssignals ansprechen, um ein Freigabesignal für die zweite Ein- und Ausgabeeinheit zu erzeugen, daß der zweiten Ein- und Ausgabeeinheit die Wiederaufnahme der Einpunkt-Datenübertragung ermöglicht.

45

50

55

60

65

6

0 009 678

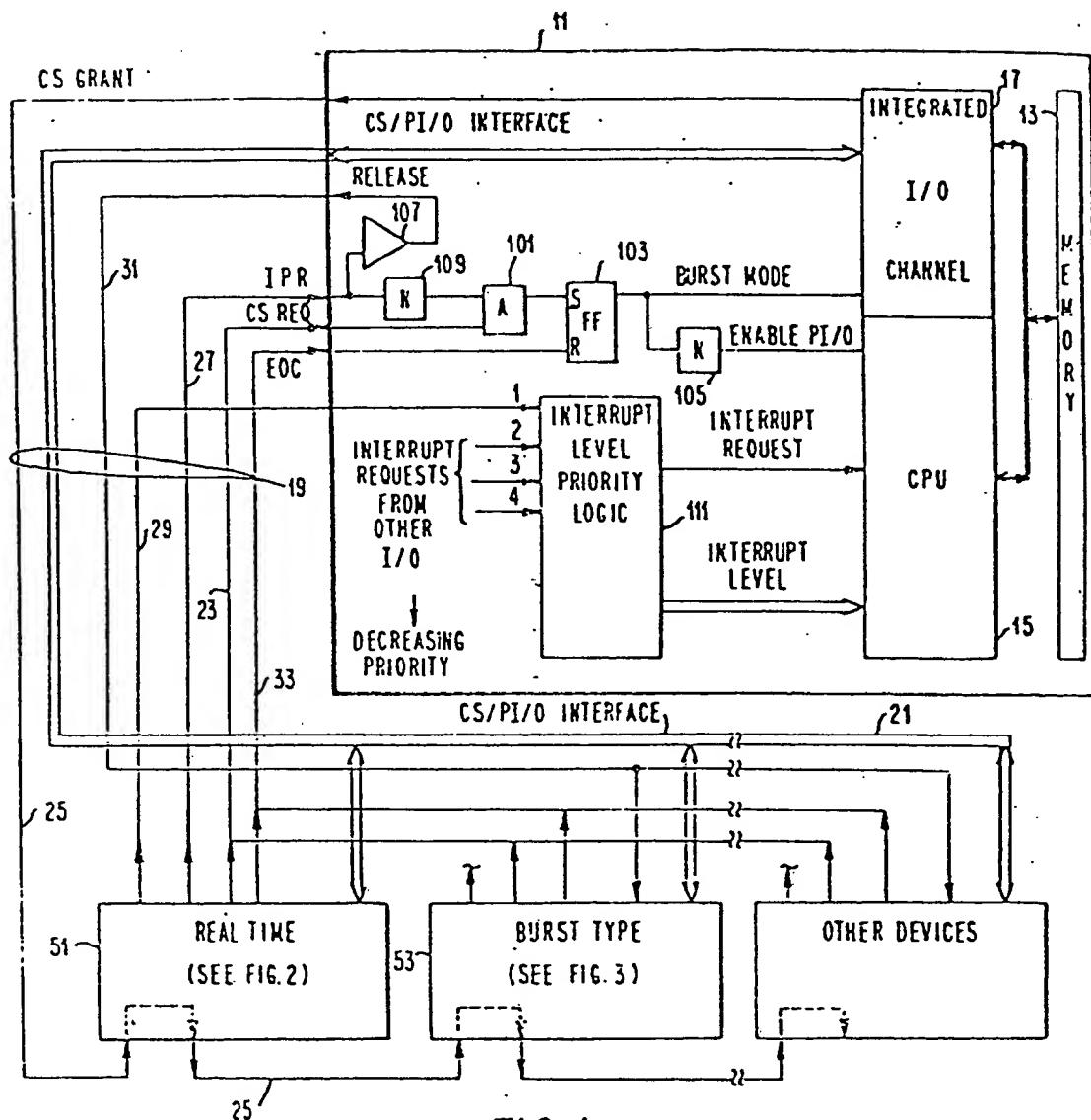


FIG. 1

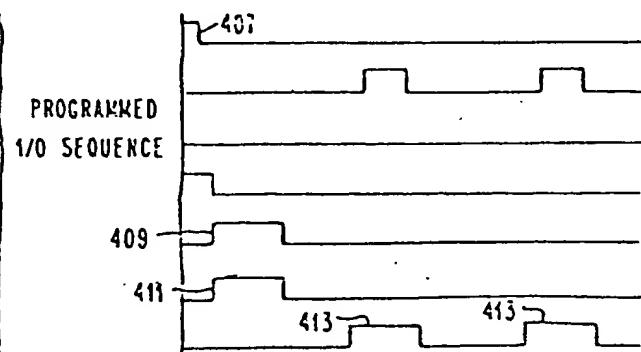
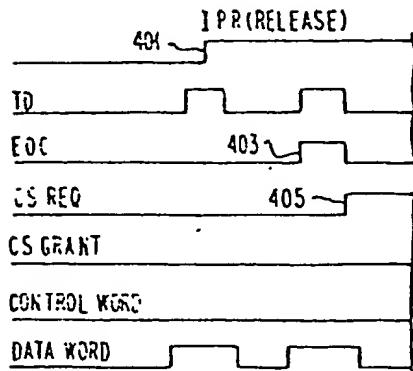


FIG. 4

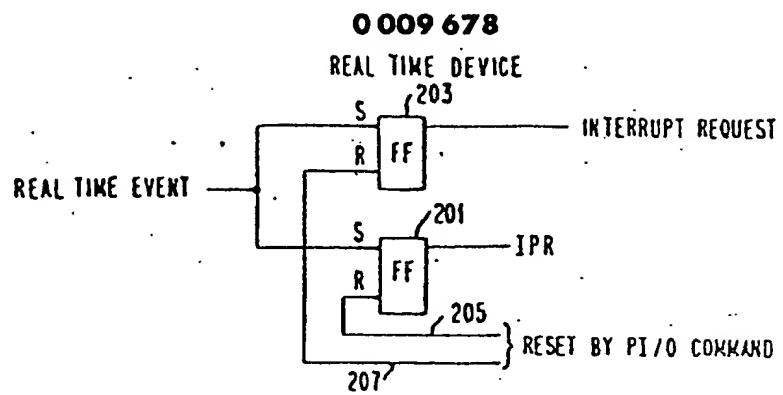


FIG. 2

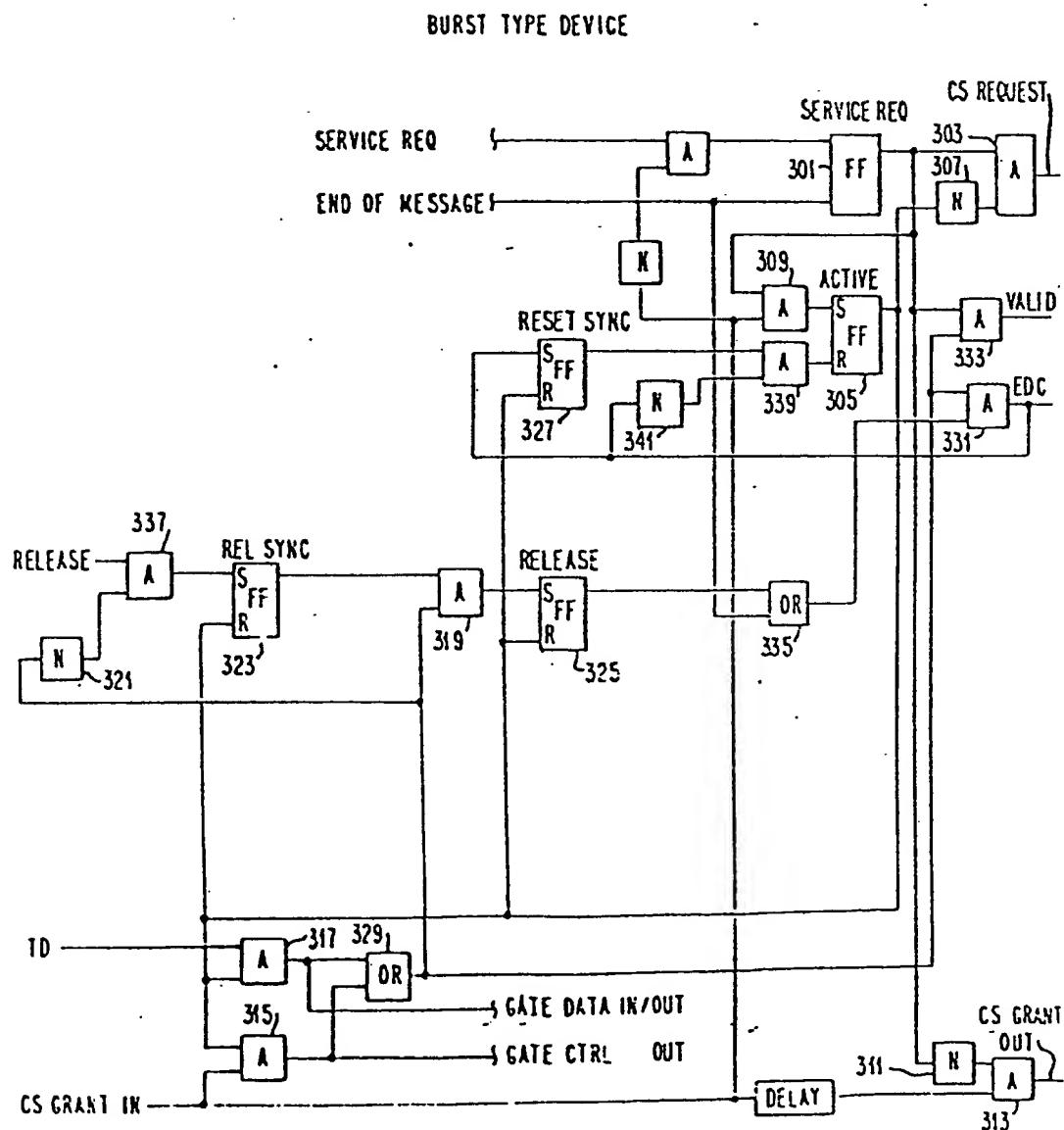


FIG. 3